Learning VB.Net

Tutorial 12 – ByVal & ByRef

Hello everyone… welcome to vb.net tutorials. These are going to be very basic tutorials about using the language to create simple applications, hope you enjoy it. If you have any notes about it, please send them to **notes@mka-soft.com** I will be happy to answer them. Finally if you find these tutorials are useful, it would be nice from you to send a small donation via PayPal to **donation@mka-soft.com**.

Tutorial posted on 2010-March-07.

ByVal & ByRef

One of the useful properties of functions is that it allows you to propagate changes in a parameter back to its original variable. For example consider this simple functions:

```
Function TestByVal(ByVal N As Integer) As Integer
    N = 0
    Return N
End Function


Function TestByRef(ByRef N As Integer) As Integer
    N = 0
    Return N
End Function
```

Now if you call the functions like this:

```
Dim K1 As Integer = 100
TestByVal(K1)
MsgBox(K1)

Dim K2 As Integer = 100
TestByRef(K2)
MsgBox(K2)
```

The first MSGBOX will display the value 100, while the second will show the value zero. This can be explained as follows:

When the computer sees the ByVal keyword it will create an independent copy of the variable you are passing to the function and work on it. In our example TestByVal, the function sets the variable N to zero. Since it is passed by value this won't affect the orginal parameter K1.

However when you use ByRef keyword the computer will create a copy of the variable that is linked to the original parameter. In other words the variable N in the example is another name for the variable K2, so when you change N in the second function, K2 changes as well. So this is why you get the value zero from the second MSGBOX.

Now let us create a useful example to swap two numbers:

```
' swap function
Function Swap(ByRef V1 As Integer, ByRef V2 As Integer)
    Dim Tmp As Integer
    Tmp = V1
    V1 = V2
    V2 = Tmp
End Function
```

Here you are passing two numbers into the function and the function swaps them. Now to call the function you can try something like this:

```
    Dim N1 As Integer
    Dim N2 As Integer
    N1 = InputBox("Enter N1")
    N2 = InputBox("Enter N2")
    Swap(N1, N2)
    MsgBox("N1:" & N1)
    MsgBox("N2:" & N2)
```

So whenever you call the function the numbers are changed. Try to modify the code of the function to have some errors, for example let it be like this:

```
' swap function modified
Function Swap(ByVal V1 As Integer, ByVal V2 As Integer)
    Dim Tmp As Integer
    Tmp = V1
    V1 = V2
    V2 = Tmp
End Function
```

Try the same code and you will see it will not work as we clarified before. Also try modifying the function to be like this:

```
' swap function modified
Function Swap(ByVal V1 As Integer, ByRef V2 As Integer)
    Dim Tmp As Integer
    Tmp = V1
    V1 = V2
    V2 = Tmp
End Function
```

Here you are going to get strange result, one of the values will be changed, while the other not simply because one of the parameters if by reference and the other is by value. So just in case you are getting unexpected results and your function code appear to be correct, just make sure that the ByVal and ByRef keywords are used correctly.

So this is all for today. If you need the source file, you can get it from the web site. Next tutorial will work with subroutine, and try to create a useful sort program. If you have notes about this tutorial, email me at: **notes@mka-soft.com**.

Thanks.

mkaatr